DOCUMENT RESUME

ED 398 882 IR 018 062

AUTHOR Cimikowski, Linda; Cook, Joan
TITLE A Model Instructional Computing Course for Preservice Teachers.
PUB DATE 96
NOTE 9p.; In: Call of the North, NECC '96. Proceedings of the Annual National Educational Computing Conference (17th, Minneapolis, Minnesota, June 11-13, 1996); see IR 018 057.
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Science Education; Computer Software Evaluation; Computer Uses in Education; Cooperative Learning; *Course Content; *Curriculum Guides; Educational Technology; Higher Education; Hypermedia; Information Technology; Introductory Courses; *Preservice Teacher Education; Required Courses; *Teacher Education Curriculum; Teaching Methods; Telecommunications
IDENTIFIERS Computer Ethics; Montana State University

ABSTRACT

Foundations of Instructional Computing is a required one-semester, two credit course in the Education Department of Montana State University (Bozeman). The major objective of the course is to help preservice teachers develop use of computer technology that is confident, thoughtful, and integrated into their individual teaching philosophy and strategies. Course content covers computer terminology, hardware selection, use of productivity tools, types of educational software, software evaluation and integration, telecommunications, use of hypermedia, computer ethics, and discussion of numerous computer technology issues related to education. The following key components of the course are discussed: objectives, lecture, text, cooperative learning, software integration and evaluation, telecommunications, homework, tests, final project, assessments, and student evaluation. (Author/SWC)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 398 882

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

- ☐ This document has been reproduced as received from the person or organization originating it.
- ☐ Minor changes have been made to improve reproduction quality.
- • Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
D. Ingham

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Paper

A Model Instructional Computing Course for Preservice Teachers

Linda Cimikowski
Montana State University
213 Reid Hall
Bozeman, MT 59717
406.994.2293
iedlc@trex.oscs.montana.edu

Joan Cook
Montana State University
213 Reid Hall
Bozeman, MT 59717
406.994.5950
iedlc@trex.oscs.montana.edu

Key Words: **preservice education, instructional computing, higher education, teaching strategies, course development**

BEST COPY AVAILABLE

IR018062

Abstract

Foundations of Instructional Computing is a required one-semester, two credit course in the Education Department of Montana State University. This course offers students a foundation in learning to effectively integrate computers into their teaching. The following key components of the class are discussed: objectives, lecture, text, cooperative learning, software integration and evaluation, telecommunications, homework, tests, final project, assessment, and student evaluation.

Introduction

Technological changes are transforming society and the ways in which we learn. Teachers need to be computer literate and prepared to use the computer effectively in their teaching. Their computer use should be confident, thoughtful, and integrated into their teaching philosophies as well as their strategies. Teachers must know how and when to use the tools of computer technology to develop and expand their students' abilities. For these reasons, the education department of Montana State University made major modifications to the Foundations of Instructional Computing (EDCI320) course in 1991. Since then, continuous refinement and experimentation has taken place in this course every semester. Foundations of Instructional Computing course components include computer terminology, hardware selection, use of productivity tools, types of educational software, software evaluation and integration, telecommunications, use of hypermedia, computer ethics, and discussion of numerous computer technology issues related to education. The major objective of this course is to help preservice teachers develop use of computer technology that is confident, thoughtful, and integrated into one's teaching philosophy and strategies. The ISTE (International Society for Technology in Education) and NCATE (National Council for Accreditation of Teacher Education) guidelines for computer/technology competencies for all teachers are used as subobjectives for the course. These guidelines are as follows:

All candidates seeking initial certification or endorsements in teacher preparation programs should receive foundations that prepare them to:

1. demonstrate ability to operate a computer system in order to successfully use software.
2. evaluate and use computers and related technologies to support the instructional process.
3. apply current instructional principles, research and appropriate assessment practices to the use of computers and related technologies.
4. explore, evaluate, and use computer/technology-based materials, including applications, educational software, and associated documentation.
5. demonstrate knowledge of uses of computers for problem solving, data collection, information management, communications, presentation, and decision making.
6. design and develop student learning activities that integrate computing and technology for a variety of student grouping strategies and for diverse student populations.
7. evaluate, select, and integrate computer/technology based instruction in the curriculum of one's subject area(s) and/or grade levels.
8. demonstrate knowledge of uses of multimedia, hypermedia, and telecommunications to support instruction.
9. demonstrate skill in using productivity tools for professional and personal use, including word processing, database, spreadsheet, and print/graphics utilities.

10. demonstrate knowledge of equity, legal, and human issues of computing and technology use as they relate to society and model appropriate behaviors.
11. identify resources for staying current in applications of computing and related technologies in education.
12. use computer-based technologies to access information to enhance personal and professional productivity.
13. apply computer and related technologies to facilitate emerging roles of the learner and the educator (The ISTE Accreditation Committee, 1993, pp. 8–9)."

In addition, Foundations of Instructional Computing was designed to model the very same innovative teaching strategies that students are expected to use in their classroom and to model the integration of technology into instruction.

Lecture

Foundations of Instructional Computing was originally composed of a one hour lecture period once per week and a two hour lab once per week. The class has drastically improved by changing the course to meet for one three hour lecture/lab session per week. Students are able to spend more time in the lab than before, thus becoming more confident in using the various technologies. Student comments indicate that they prefer the lecture being incorporated into the lab because they like being able to immediately transfer knowledge from the lecture into hands-on activities in the lab.

Lectures have been incorporated as a part of the lab experience and have been reduced to as little time as possible. Electronic presentations are used during lecture so that the students have visual as well as auditory input. Each lab instructor is encouraged to ask questions to stimulate discussions during the lecture to facilitate constructive learning. Students are also encouraged to take notes using word processing software.

A few of the lectures (e.g. teaching and learning theory, test creation, and telecommunications) have been incorporated into *HyperCard* stacks. This provides a radically different way for the students to learn the material and reduces intellectual passivity and linear thinking, common problems associated with the traditional lecture method of teaching. The students later learn *HyperCard* basics themselves and use *HyperCard* to design a piece of educational software that will be incorporated into their final unit.

Text

Student dissatisfaction with the several different texts that have been used with EDCI320 has led to a new strategy for gathering and disseminating necessary information. Instead of using a set text, a variety of texts are put on reserve in the library, and each student has readings on a particular topic. Each student is responsible for reading his/her material, gleaning the most important points, and then being the guest teacher who will teach the information to the remainder of the class. Students are encouraged to use creative teaching strategies. If students decide to use lecture as their teaching strategy, they are encouraged to use *ClarisWorks* Word Processing slide show or *PowerPoint* for creating presentations to use with the lecture. Variations of use of having the students teach the information in the past have revealed that students must be limited to a short presentation (five minutes is currently being tried) and must be instructed to only teach the most important and relevant parts. Otherwise, some students tend to make their presentations too long and complex. In order to incorporate technology into this exercise, each student is required to learn to use *ClarisWorks* Draw to create an attractive newsletter complete with columns and graphics. The contents of the newsletter is the important information gleaned from the reading assignment. The

student then distributes the newsletter during his/her presentation, and it can be used as a study guide for tests.

Students in EDCI320 are also required to review available computer magazines, such as *Learning and Leading With Technology* (ISTE Journal of Educational Technology Practice and Policy), *Connections, Journal of Research on Computing in Education*, *T.H.E.* (Technological Horizons in Education Journal), and *Technology and Learning*. They must then select an article of interest to themselves and to their classmates and give a short presentation on what they have learned. This is an excellent opportunity for both the students and instructors to share the most recent instructional technological news. In addition, students become familiar with some of the journals that will keep them, as teachers, technologically current. If limited class time prevents oral reports, then the highlights of the journal articles could be placed in a class newsgroup on the university network, thus giving additional experience in using the network while creating interesting reading.

Cooperative Learning

Cooperative learning is used to replace and/or supplement certain lectures. Cooperative learning gives students a more active, analytical role and encourages them to take responsibility for their own learning. Instead of sitting passively at their desks taking notes during lecture, the students are able to spend time thinking, synthesizing information, and applying ideas.

During the piracy/ethics/copyright portion students are placed in cooperative groups to discuss ethical dilemmas common to teachers using technology. Discussion is usually fervent and inevitably leads to the students convincing each other, with little instructor intervention, that teachers must practice ethical procedures regarding piracy, copyright violation, etc.

A cooperative group activity is also used to demonstrate the use of stations in a classroom. Sending e-mail, subscribing and posting to newsgroups, using a modem, and using Apple IIes is taught simultaneously using cooperative groups. Groups rotate through stations working together to make sure that each member of the group understands the concepts or skills at each station. Variations of this teaching model are discussed so students understand that even with one computer in a classroom there are many teaching/learning situations that can take place.

Students also work in groups of two or three on a computer to perform creative thinking activities, to learn various pieces of software, and to design and produce materials. Each student is usually assigned a responsibility, with one student (the reporter) inputting the information on the keyboard, one student (the encourager) making sure each person is participating and receiving encouragement, and one student (the checker) making sure that each team member understands what is being discussed and accomplished (Johnson, Johnson, & Smith, 1991, pp. 3:3–3:6). One of the student's favorite cooperative activities is the introduction to divergent thinking and database relationships. Each student group lists three nouns using word processing. Next, each group decides in what way each of the three nouns are like a piece of toast. This leads to a discussion of divergent versus convergent thinking and to a discussion about how data in a database can be categorized or related.

Software Integration and Evaluation

One of the most important elements of Foundations of Instructional Computing is software evaluation. Three weeks are used for exploring and learning about software and software evaluation, while all other topics are usually limited to one week. Exposure to as many different types of software as possible is emphasized, especially

since once these students become teachers or return to the classroom, far too little time is available for good software evaluation and reflection on its use in the curriculum.

Students work in groups or individually and are encouraged to evaluate software in their own area. They use evaluation sheets actually used in the local schools so that the experience is more meaningful. They are encouraged to find software that will fit into their final unit, which focuses on a topic in their own area of study and that will be useful to them when they are actually teaching. In order to assure that the students evaluate all major types of computer-assisted instruction software and that they use it to the greatest advantage, they are given the following directions:

1. Use a piece of drill and practice software to refresh some skill you previously learned. Analyze the program to see if any of the following elements are given consideration: speed can be adjusted, operation is easily learned, positive reinforcement is more effective than negative reinforcement, content is accurate.
2. Use a tutorial to learn something new. Write a short description of the program including your likes and dislikes and a specific example of how you could use it in the classroom. How could it be improved? A brief list of characteristics is given so students know what to look for in judging the performance:
    - Does it have a clear objective?
    - Does the program allow students to redo missed answers?
    - Does it provide motivation to learn?
    - Does it provide feedback on the student's performance?
3. Evaluate a problem-solving software, a simulation software, and a CD-ROM (the CD-ROM can be any type). Use the form used in the local schools to evaluate the software.

Material from other classes taught in the Education Department is incorporated into the software evaluation portion of the class. Students must incorporate reviewed software into lesson plans that follow the lesson plan format learned in methods classes. They integrate their knowledge of Bloom's taxonomy, learned in Educational Psychology, into writing objectives for the lesson plans. They integrate their knowledge of Piaget's developmental stages, learned in Educational Psychology, into learning how instructional software can be used employing different strategies to reach students at different developmental stages. This particular concept is taught using *Building Perspectives*, a problem-solving piece of software. Using *Building Perspectives*, students look at nine high-rise buildings from an aerial view, front view, right view, left view, and back view. They must determine how many floors there are in each of the nine high-rise buildings. Some students try doing the exercise with no physical help, while some are given building blocks so they can physically create the buildings while looking at the different views. After the students have used this piece of software with and without building blocks, they see how a teacher can use manipulatives to bring an abstract concept to a more concrete level if needed. Students also use their knowledge of different learning styles, learned in Educational Psychology, to study how different teaching strategies can be used with a piece of software so that it is effective with students with different learning styles. As a part of this exercise, students are required to complete the "MAX Inventory of Learning Styles" on the computer. Each student is asked to bring a printout of his/her "brain," which rates how he/she scored on each of the learning styles designed by Howard Gardner. This exercise clarifies for the students how different each person can be and how important it is for teachers to use a variety of teaching strategies with a variety of software so that all students can effectively learn.

Telecommunications

The university telecommunications network is a technology that has provided a much needed aspect of EDCI320. Students must set up an account and send a message to their lab instructor describing an idea they would use in their own classroom using telecommunications. During the course students are encouraged to provide input to their lab instructors using e-mail.

In addition, the class has established its own newsgroup, where both students and instructors can post class announcements, discuss thought-provoking questions, and reflect on the effectiveness of teaching strategies used. When a teacher or student posts a message to the newsgroup, the message is listed in the newsgroup for all subscribers to see and to respond to with their own comments.

The class also uses *Netscape* software to access the World Wide Web. Students are easily motivated to use *Netscape* because it is easy to use and they love the colorful graphics. Students must locate lesson plans and/or information that they could use in their own classrooms in the future and that they can use in their final project.

Homework

Homework provides an excellent opportunity for incorporating the use of technology. Students must do all their homework using a computer. They must keep copies of all homework on an original disk and a backup disk so that no questions arise as to whether the student has done all the work if a homework is accidentally not recorded. Students can redo many of the assignments, which assures that they have every opportunity to properly learn to use the technology (e.g. *ClarisWorks* Word Processing, *ClarisWorks* Spreadsheet, *ClarisWorks* Database, *ClarisWorks* Draw, *MakeTest*, *SuperPaint*, and the World Wide Web) and have an understanding of the educational issues. Saving the work on a disk makes it wonderfully simple for the student to redo the work to make it right. Students also keep records of their own homework and test scores in a spreadsheet so they can track their grades.

Tests

A technology Bingo game is played as a review the week before the test. Bingo software is used to create Bingo boards that contain technology questions that might appear on the test. Each class member is given a Bingo board and must circulate around the classroom having other students answer the questions. The student who knows the answer must explain it fully and then sign his/her name in the bingo square containing that question. The first student who has a total blackout receives a free disk. Even college students find this exercise exciting, so this portion of class is always extremely lively.

Tests are used to reinforce student technology skills and knowledge, plus innovative strategies are used to ensure the students learn the answers to the questions they missed. Quizzes and the essay portion of tests must be done on the word processor. Essay answers must be free of spelling errors (since spell-check can be used), and the students demonstrate their mastery of using the lab printers when they print. After the students have received their graded papers, they can research the correct answer and use the word processor to write full paragraphs discussing the answers to true-false, multiple choice, and fill-in questions in order to recoup half the points they missed. This assures that they learn information vital to using computers in the classroom. An alternative testing method that has been tried in this class is allowing small groups to take the first part of the test (true-false, multiple-choice, fill-in) and requiring each individual to do the essay portion alone. If students are not happy with the group score, they can

individually research each missed question and write up the answers to recoup the missed points.

Final Project

All in-class work and out-of-class assignments lead toward the creation of a final project that ties together all the students have learned. Students select a unit topic that they will be able to actually use in their classroom in the future. They then create a portfolio of all the portions of that unit that relate to computers. Following is a list of what currently must be included in the final unit:

- a sign or poster created using *PrintShop* or *SuperPaint* to introduce the unit (this must list the main unit objectives).
- a lesson plan describing how they plan to incorporate their *HyperCard* stack they created.
- a second lesson plan incorporating the piece of CAI software they plan to use in their unit.
- a completed evaluation form evaluating the piece of software in the preceding lesson plan.
- a newsletter that could be used as a study guide to reinforce the objectives of the unit.
- a printout of a relevant lesson plan or other information that was retrieved using the World Wide Web.
- a description of an activity that uses a spreadsheet or database as a teaching device for the selected subject matter and a sample of the completed spreadsheet or database.
- a test created using *MakeTest* that would be used to test learning of the objectives of the unit.
- a mail merged letter to the parents of several students in the class describing the unit and what the teacher expects of the student.
- a printout from a piece of tool software used to enhance the unit (e.g. crossword puzzle, certificate, etc.).

Assessment

Student assessment is based upon quiz and test grades, quality of lab assignments, participation in group activities, the oral report on the student's journal articles, quality of the student's guest teaching of his/her assigned chapters to the class, and quality of the final unit project. Each activity is assigned point values. These point values are then used to compute the students' final grades.

Student Evaluation of Course

During the last class session, students are asked to complete course evaluations. These evaluations are read and used to assist in determining changes to be made for the next semester's class.

Overall, the evaluations are extremely positive for this course. Students especially enjoy the hands-on lab activities. Students comment continually on how relevant and important the material is to them as future teachers. Since the instructors also believe in the relevance and importance of the material and this is reflected in their teaching, the students are continually commenting on the enthusiasm of the instructors.

Evaluations of the particular class format described in this paper revealed that students preferred to return to regular text reading assignments. Many students felt that the time allocated for students to "guest teach" their assigned text chapters used too much valuable computer lab time, and they desired to return regular text readings. Students also requested that even more time be spent in the computer lab learning telecommunication skills, especially use of *Netscape* and the World Wide Web.

While graduating students are not officially interviewed to determine how this course has assisted them in their teaching, many graduates who have obtained teaching jobs do return to give the instructors input. The general returning graduate comments indicate that this class has proven to be one of their more valuable courses. Returning graduates iterate that they use the skills learned in this Foundations of Instructional Computing course frequently in their classrooms. Graduates often tell us that they particularly use newsletters in their classroom and are extremely happy that newsletters were covered in this course. A few graduates have informed us that they have become the technology coordinator of their school and that they directly attribute this to having taken this course.

Conclusion

For many teacher preparation programs, such as ours, only one basic instructional computing course is required. This is blatantly an inadequate technological preparation for future teachers with the current accelerating technological advances that are occurring. Each of us, therefore, must work constantly to improve these foundation courses. They must teach the basic survival information and provide a foundation for further life-long learning, while they must model the very things that they teach. Sharing how and what is taught in these classes is a beginning for making all of our classes even more stimulating and productive.

References

ISTE Accreditation Committee. (1993). *Curriculum Guidelines for Accreditation of Educational Computing and Technology Programs, 2nd Edition*. USA: International Society for Technology in Education.

Johnson, D.W., Johnson, Roger, T., & Smith, K.A. (1991). *Active Learning: Cooperation in the College Classroom*. Edina, MN: Interaction Book Co.